

Testimony - Release Notes

2.50 — Last update: 6 September 2022

Basis Technologies

Table of Contents

- Introduction 1
- SAP Certification 2
- General Enhancements 3
 - Configuration Parameters 4
 - Check steps 5
 - Playback system parameters 6
 - User Preference Configuration 7
 - Exclusion Configuration 9
 - Enhancements 10
- Key Enhancements..... 11
 - Screen Field Validation 12
 - Predictive Difference Analyzer 13
 - Advanced Tree and Table Control Selection 14
 - Defect Sensitivity 15
 - SLG Enhancements 16
 - Recording Extractor 17
 - Defect Flow Report 18
 - Report Launcher 19
 - Root Cause Analysis 20
- Other Product Changes 21
- Notes on S/4 HANA Systems 23
- Bug Fixes 24

Introduction

Testimony v2.50 was released in December 2021.

Previous releases of Testimony are detailed in separate Release Notes:

- **Testimony v2.40** (Released July 2021)
- **Testimony v2.30** (Released November 2020)
- **Testimony v2.21** (Released March 2020)
- **Testimony v2.20** (Released November 2019)
- **Testimony v2.13** (Released June 2019)
- **Testimony v2.12** (Released May 2019)
- **Testimony v2.11** (Released April 2019)
- **Testimony v2.10** (Released February 2019)
- **Testimony v2.09** (Released January 2019)
- **Testimony v2.08** (Released December 2018)
- **Testimony v2.07** (Released November 2018)
- **Testimony v2.06** (Released October 2018) – **Interim Release Only**
- **Testimony v2.05** (Released October 2018)
- **Testimony v2.04** (Released October 2018)
- **Testimony v2.03** (Released September 2018) – **Interim Release Only**
- **Testimony v2.02** (Released September 2018) – **Interim Release Only**
- **Testimony v2.01** (Released September 2018) – **Interim Release Only**
- **Testimony v2.00** (Released August 2018)

SAP Certification

Testimony is a SAP certified solution:

- Certified for deployment on SAP NetWeaver 7.50 via the SAP integration scenario ABAP Add-On Deployment for SAP NetWeaver (**SAP report 12649**)
- Certified for deployment on SAP S/4HANA 1709 via the SAP integration scenario ABAP Add-On Deployment for SAP S/4HANA (**SAP report 12657**)
- SAP Solution Manager Ready functionality

All Testimony SAP components exist within Basis Technologies' own namespace **/BTI/**.

General Enhancements

This section describes the general enhancements that have been delivered in Testimony v2.50.

The key areas described in this section are:

- **Configuration parameters** – New parameters added in this release
- **Check steps** – This includes new check steps that have been introduced in this release including enhancements to existing check steps
- **Playback system parameters** – Any playback system parameters that have been identified for importance for playbacks
- **User preference configuration** – New default user preference configuration
- **Exclusion configuration** – Describes the new entries added to the default filter set containing defaulted exclusion entries
- **Enhancements** – Changes to the enhancements in both recording and playback

Configuration Parameters

A number of new parameters have been added in this release of Testimony:

Parameter Name	Area	Description
TARGSYS_DEBUG_LOG	Playback	Allows logging during the playback process in order to investigate particular issues that have been experienced previously (primarily whether playback enhancements were active or not)
FORCE_BASELINE_SEQ	Playback	This parameter is only used in the context of double playback. Please set it to true in order to ensure that when the second queue is executed, it will base the sequence using the execution queue step start/end timestamps of the base-line queue
FORCE_BASELIN_RESULT	Playback	This parameter is only used in the context of double playback and the previous parameter for forcing the baseline queue sequence must also be activated. It will ensure that the same results (failure or error) is enforced in the second queue based on what was seen in the baseline queue.

Check steps

The following check steps have been created or changed in this release:

Check Step	Phase	Type	Description
Fill number range buffers	Recording	Preparation	This preparation step gets the recording ready when a large number of application servers are involved in the system being recorded. The number range for storing each “transaction” is extensively buffered, but if this is the first time a recording is activated, then the number range will be overloaded on recording activation since each application server will request immediately their allocation of buffered numbers. This step pre-loads these buffers prior to turning on the recording.
Check PDA has been executed	Playback	Check	Prior to a playback being started, this check will ensure that Predictive Difference Analyzer (PDA) has been executed to proactively check for any obvious errors or failures that might occur. If it has not been run yet on the execution queue, then this will flag up a failure in this check step.

Playback system parameters

No profile parameters have been added or changed during the playback check step in 2.50.

User Preference Configuration

The following user preference configuration has been added or amended to the configuration tables in order to more accurately capture this data during the recording and restore during the playback. Please review these entries and check if any custom configuration entries should be added (for example for Y/Z transactions that store and use user-specific settings).

Transaction	Configuration Table
MB51	ESDUS
ME21N	ESDUS
ME22N	ESDUS
ME23N	ESDUS
ME29N	ESDUS
ME51N	ESDUS
ME52N	ESDUS
ME53N	ESDUS
MEPO	ESDUS
MIGO	ESDUS
MIRO	ESDUS
PA30	LTDX
PPO1	LTDX
PPO2	LTDX
PPO3	LTDX
PPO4	LTDX
PPO5	LTDX
PPO6	LTDX
PPOC	LTDX
PPOCA	LTDX
PPOCE	LTDX
PPOCW	LTDX
PPOC_OLD	LTDX

PPOM	LTDX
PPOMA	LTDX
PPOME	LTDX
PPOMW	LTDX
PPOM_OLD	LTDX
PPOS	LTDX
PPOSA	LTDX
PPOSE	LTDX
PPOSW	LTDX
PPOS_OLD	LTDX



Important: All user preferences are configured in the central system only. The two important tables are **/BTI/AUT_C_USPA** and **/BTI/AUT_C_USPR**.

Exclusion Configuration

The default values delivered in v2.50 for the exclusion configuration in Filter Sets have been enhanced.

There are now approximately **250 objects** in the default configuration for exclusions with batch jobs, dialog transactions, inbound RFC's and inbound web-services. Please review these entries prior to starting the process and ensure that you agree on the configured entries.



Don't forget you can always over-ride the standard delivered filter set and configure your own. This can also be done after reviewing the usage data from transaction ST03 or STAD. However, you can also do this configuration after a recording in the "Performance Analysis" area of Testimony.

Enhancements

Recording Enhancements

The following changes have been done with regards to the recording enhancements available in Testimony 2.50.

1. **Capture BDC QID** – This is a new enhancement in 2.50 but it is currently under review and is not released.
2. **Virtualization of TEXT_CONVERT_XLS_TO_SAP** – This is a standard function module in ECC which is captured during the recording.
3. **Simple ALV tree: Adv. Row Sel. Add nodes** – Used for capturing the data contained within node selection for tree controls.
4. **Simple ALV tree: Adv. Row Sel. Update** – Used for capturing the data contained within node selection for tree controls.

Playback Enhancements

The following changes have been done with regards to the recording enhancements available in Testimony 2.50.

1. **Capture BDC QID** – This is a new enhancement in 2.50 but it is currently under review and is not released.
2. **ALSM Excelt to Internal Table** – Used for virtualizing a standard ECC function module.
3. **Virtualization of TEXT_CONVERT_XLS_TO_SAP** – This is a standard function module in ECC which is virtualized during the playback.

Key Enhancements

The key areas that have been enhanced in v2.50 are listed below:

- **Screen Field Validation** – Capture and validate screen fields during recording and playback
- **Predictive Difference Analyzer** – Ensure source and target system objects are the same
- **Advanced Tree Node Selection** – Support for tree control node selection based upon tree content
- **Advanced Table Control Selection** – Support for table control row selection based upon content
- **Defect Sensitivity** – Ability to prioritise defects automatically based upon custom criteria
- **SLG enhancements** – Enable inbound RFC and batch jobs to be supported by SLG
- **Recording Extractor** – Extract data captured in the recording for a separate use-case
- **Defect Flow Report** – Extract data for high level analysis of the playback results
- **Report Launcher** – Visibility of many internal reports built over the years but now forgotten
- **Root Cause Analysis** – Useability improvements and internal validation of capability

Screen Field Validation

This new functionality in 2.50 allows a customer to capture specifically configured SAP GUI dialog screen fields during the recording and validate that these fields contain the same values during the playback.

The original requirement for this came from a pharmaceutical company (FDA regulated) who needed to ensure that the “expiry date” for a batch of drugs that had been produced matched what was expected. This was one of their critical business processes that must be tested as a part of FDA compliance.

In order to capture screen fields during the recording, you must configure the table **/BTI/AUT_C_DVFL** within the central system that specifies which field(s) upon which screens within which transactions should be captured and validated. You perform this configuration via transaction SM30.



Screen field capture and validation is “**white-listed**”. This means that, unless you configure this table, no screen fields will be captured during the recording.



You are able to configure this table with wild-cards for the various fields. So if you would like to capture all fields for a given transaction code and screen number, then you are able to do this. Equally, you may capture a certain screen field (upon a given program and screen number) regardless of the transaction code that initiated it.

Additional information for configuring this new functionality can be found in this [FAQ](#) in Zendesk.



The key table for configuration of this functionality is **/BTI/AUT_C_DVFL**.

Predictive Difference Analyzer

When critical differences exist in the target system which cause either mass technical errors or mass failures in the playback, it would be better to be more proactive regarding this rather than reactive when it is too late. Before a playback is started, Predictive Difference Analyzer (PDA) will review any differences between all relevant objects in the **source system** and those same objects in the **target system**.

Any “differences” between the objects compared are flagged up to the operator and are marked as either:

- **Errors** – Difference in target will cause issues during the playback
- **Warnings** – Differences to be aware of but not necessarily playback issues

The process for utilizing PDA is as follows:

1. Navigate to PDA execution drawer item
2. Click on the “Prepare” menu item (single threaded for small execution queues, Diffuser version for large queues)
3. Run the preparation program for the execution queue (new run should appear)
4. Select the run and choose “Compare” menu item. Single threaded or Diffuser option
5. Once complete, double click the PDA run that has been analyzed and compared
6. You can view a summary of what has been detected by PDA by object type. Alternatively, you can view a summary of all differences detected by the “failure reason” (tab 2). The last tab shows the results of PDA by individual object. Filters are available in all grids.
7. For errors (or warnings) that are found, you can drill down to a list of these which provides additional information and further help or documentation. From there remediation can be triggered as well.

Remediation

It is possible to remediate the content (input/output container elements) of test scripts where PDA has detected a scenario which will cause errors or failures. This functionality only supports adding or replacing fields within containers (with default values) for dialog transactions and inbound RFC. It is triggered from remediation option when reviewing the results of the comparison

✿ **Example:** New mandatory parameter added to an RFC in the target
In this example, if the playback were to run then every test script containing this RFC would return a technical error. Since PDA detects this difference, you can run remediation to update all related test script steps and add this new parameter to the input container with a default value that you can provide

Advanced Tree and Table Control Selection

In 2.30 / 2.40 we introduced advanced row selection for ALV grids and ABAP lists. In v2.50 this has been extended to support table controls and ALV tree controls which are the final remaining gap in SAP technology for selecting rows/nodes in the most advanced manner.

In earlier versions of Testimony, just the “index” of the row (or node) was recorded and used for the playback. For example, if row 10 of an ALV grid had been selected by a user in the recording, then during the playback the bot would simply select row 10 (regardless of whether the data within the grid is the same during the playback). Advanced row selection changes all of this by selecting the row that contains the same data as what was in the recording. This functionality has been extended to table controls and tree controls in v2.50.

There are two new enhancements that must be enabled during the recording:

1. Table Controls – GRSC, GRSR and OUPF enhancements must be activated
2. Tree Controls – TRRS, TRSS and TRSU enhancements must be activated

✳ Row record linkage types are also generated for these controls in the same way this is done for ALV grids and ABAP lists. You can view the row record (and the data contained within it) via the linkage type of the same name.

Failure reasons are the same as ALV grids and ABAP lists when the row or tree node cannot be found.

Defect Sensitivity

Currently Testimony can filter and work defects based on their priority, i.e. Critical, High, Medium, Low. The problem with this is there could be many Defects with the same priority and we have no way of knowing which of these are the most important ones.

Defect Sensitivity introduces a new metric called 'Impact Score' that allows the customer to allocate a score for a defect based upon various criteria. This 'Impact Score' is automatically calculated (if "Perform Impact Calculation" is turned on in the propose defects program). Default configuration is provided with 2.50.

The possible categories that can be used to calculate the 'Impact Score' are:

- Actual Message Type
- Component
- Failure Reason
- Object Type
- Object Pass Rate
- Priority
- Root Cause
- Scripts in Defect as a % of the Object
- Number of Scripts in Defect
- Object
- Number of Execution Queues
- Number of Unique Users
- Actual Message ID / Message Number
- Next Program / Screen
- System ID

The following tables are used in the configuration of defect sensitivity:

Defect Impact Weighting – /BTI/AUT_C_BTIW

Allows you to activate/deactivate a Category for the Impact Score calculation. Allows you to assign a weighting to a Category, by default they all ship with 1

Defect Impact Scoring – /BTI/AUT_C_BTIS

Allows you to assign Scores to Category Values

Defect Impact Value – /BTI/AUT_C_BTIV

Allows you to convert the Overall Impact Score into a 'Textual' value.



Once you have calculated a defect sensitivity impact score for your defects, you can then see them within the defect management grid. You are then able to filter and sort when displaying the list of defects.

SLG Enhancements

This is more for internal purposes at Basis Technologies for testing of Testimony. However, some customers have requested the use of SLG (Synthetic Load Generation) for their own use on-site.

SLG has been enhanced to allow test scripts to be setup for **inbound RFC** and **batch jobs**.

Recording Extractor

Key Information

The recording extractor allows you to perform a recording and then extract the information in the recording for further processing (potentially outside of Testimony and for purposes other than testing).

The technical names of the report are **/BTI/AUT_RECORDING_EXTRACTOR** accessed via transaction code **/BTI/AUT_REXT**.

The output of this report will generate a file that is created locally in XML format. This file can then be provided to other systems for other purposes. You can also

Defect Flow Report

The defect flow report is a tool which allows the customer to visualize the results of a playback. The output is broken down into a flow of numbers starting from the test scripts based upon their type (dialog transactions, inbound RFC etc), then those scripts that become failures or errors during the playback.

The failures are then broken down further into the various reasons for the failures and ultimately the defects they become. Further, the errors are broken down into the grouped reason for the technical error.

✿ Errors do not become defects – only failures do.

The report to run to visualize this data is program **/BTI/AUT_DEFECT_FLOW**. You specify an execution queue as the primary input parameter. Additional parameters exist, but the next most important parameter is the output file. This will download a file to your local machine so that it can be opened, copied to the clipboard and then pasted into the display of the visualization tool “Sankey Builder”.

Navigate in your browser to www.sankeymatic.com/build. Copy and paste the output of the report into the area and the sankey diagram will appear. You can also change the settings of the output of the sankey diagram (e.g. making the view larger if desired).

✿ Please note that you do not need to specify a local file to download to. If you don't, then the data will be output to the screen. In this case, you could copy the list into the clipboard and then paste into the visualization tool.

Report Launcher

Many reports have been built in Testimony over the years to support investigation of recordings and playbacks. These reports are generally executed via transaction SE38 (or they have an underlying transaction code associated with them). However, it is difficult to remember all of the various reports, what they do and how they can help with the situation you are facing.

Testimony v2.50 introduces a new screen to centralize these reports into one list (the “Report Launcher”) and allows you to trigger the execution of these reports. Sometimes these reports must be run on the source or target systems. This is pre-configured in the report launcher and you can simply click to run the report in that system.



There are approximately 50 reports pre-configured in the report launcher. You can filter them into the area they belong (e.g. playback, recording). A help option allows you to read about what the report does and how it works. You can trigger the report to run directly from the report launcher. The vast majority of the reports run upon the central system.

Root Cause Analysis

Root Cause Analysis was introduced in an earlier release of Testimony (v2.30). Further changes to the useability of the report and how it is executed have been delivered in v2.50.

The confirmed process is the following:

1. Build Execution Queue
2. Perform Playback (double playback)
3. Trigger Deep Impact Link Creation in target for “failed” objects (or other statuses)
Trigger from central system in UI in Testimony
Perform to N levels deep (see statistics on next slide)
Monitor and trigger further levels (if time permits)
4. Create “Build List” from central system representing “changes under test” in target
5. Trigger Root Cause Analysis in central finding links from failed objects (or defects) to transport requests (in the build list)
6. View RCA results at defect level – filter for “RCA Flag” in defect results screen



AC version 8.31 is the minimum required version (only in target system).

Other Product Changes

The following additional changes have been made in the Testimony v2.50 release.

Testimony Functionality	Enhancement Description
BDC Queue ID Linkage Type	Supports BDC session t-code SM35 to run correct BDC batch
Handling Dynamic ID within the recording (rather than execution)	Early stages – BDC Queue ID makes use of this. Will expand out to all linkage types in the next release
Bot “Connection Broken” popups no longer blocking the playback	Specific blocking pop-up removal now done by the bot to remove unstable network popups for disconnection from the target
Editable ALV grids now support advanced row selection	Finds the correct row in editable ALV grids based upon the contents of the data captured in the recording
Parallel processing of test plan deletion program	Provides faster archiving/deletion of test plans within the Diffuser framework
Business Transaction Storage Analysis Report	Understand the storage space for inputs, outputs and linkages across all business transactions to find large objects
Execution Queue Storage Analysis Report	Understand the storage space for inputs, outputs and linkages across test scripts and steps
Additional screen information is now captured in info popups	Captures screen data in popup informational messages (screen SAPMSDYP 0010) to support investigation of failures
Editable ALV Dynamic ID support	Finds ID's entered into the editable ALV data and replaces them if dynamic ID linkages are found
Version information screens now show correct data	Fix the display of version data when navigation to the “Version Information” screen
Support for dynamic ID within table controls	Extensions to table controls to allow identifiers to be changed during the playback
TE1 transports for ECC specific enhancements now on P25 request	For consistency, the ECC/S4 specific additional transports are now prefixed with P25K (rather than TE1K)
Playback enhancement activation logging is now available	If there are issues where playback enhancements look to not be enabled, logging is now possible to check on this
Additional enhancements to discovery mode functionality	After initial release of functionality, this was enhanced to incorporate feedback after further refinement
Inbound RFC exception handling	Additional handling of edge case exceptions during inbound RFC
Discovery mode functionality in separate and main deliverable	This functionality is now available as both a separate transport (prior to Testimony deployment) or within the main transport

Option to only delete screenshots for passed test scripts	Most of the space used up within a test plan is with the execution queue step screen-shots. You can just delete that for passed step
System user check during transfer	Removes system users during transfer to repository
Playback remote control now supports inbound web-services	The final object type (4 of 4 – excluding Fiori) is now supported with playback remote control
qRFC and tRFC units of work exception handling improved	Colgate example – performance issue around large number of queues. Plus resolved an issue with the comparison in the failed queue

Notes on S/4 HANA Systems

Testimony is built to function on a central system which records one or more source systems and plays back that recording into one or more target systems. These 3 system types would typically be separate systems. All of the systems must be ABAP stack and on Netweaver release 7.01 and above.

In the case of S/4 HANA systems (1610, 1709, 1809 and 1909), it is possible to install Testimony for recording and playback purposes. The central system would traditionally be a separate Solution Manager system.



It is important to understand that this process is not just for S/4 HANA systems. It is for any SAP system that has a Netweaver kernel of 7.53 and above. This could be relevant to ECC systems.

Before you record your S/4 HANA (or NW 7.53+ kernel) system, please liaise with Basis Technologies who will provide you with an additional transport request that should be applied to both your central system and your source S/4 HANA system. The same transport should then be applied to your S/4 HANA playback system prior to starting a playback.

This transport should be applied on top of the main recording transport request into that system.



It is critical, that you only apply this transport out of normal system hours. Please work directly with your main point of contact at Basis Technologies in order to seamlessly deploy this aspect of Testimony.

Bug Fixes

The following bugs have been addressed in the Testimony v2.50 release. Please note that this is not an exhaustive list of bug fixes, it is purely intended as a summary of the main issues reported by existing customers in the most commonly used areas of the tool.

Testimony Functionality	Bug that has been fixed
RFC parameter comparison now supports type P	Common issue with RFC's containing type P (numeric with decimals) that was causing false negatives
Double warning messages being output during playback	This was intermittently happening due to screen resizes between steps within scripts causing unexpected failures
Store and restore system variables within key enhancements	An obvious one, but it didn't materialize until 5 years after inception at Colgate (custom code)